WOOD, HERRON & EVANS, L.L.P.

BRUCE TITTEL
DAVID S. STALLARD
J. ROBERT CHAMBERS
GREGORY J. LUNN
KURT L. GROSSMAN
CLEMENT H. LUKEN, JR.
THOMAS J. BURGER
GREGORY F. AHRENS
WAYNE L. JACOBS
KURT A. SUMME
KEVIN G. ROONEY
KEITH R. HAUPT
THEODORE R. REMAKLUS
THOMAS W. HUMPHREY
SCOTT A. STINEBRUNER
DAVID H. BRINKMAN
BEVERLY A. LYMAN, PH.D.
KRISTI L. DAVIDSON
KATHRYN E. SMITH
P. ANDREW BLATT, Ph.D.
DAVID E. JEFFERIES

2700 CAREW TOWER
441 VINE STREET
CINCINNATI, OHIO 45202-2917
TELEPHONE: 513-241-2324
FACSIMILE: 513-241-6234
WEBSITE:: www.whepatent.com

PATENT, TRADEMARK, COPYRIGHT
AND UNFAIR COMPETITION LAW
AND RELATED LITIGATION

EDMUND P. WOOD 1923-1968
TRUMAN A. HERRON 1935-1976
EDWARD B. EVANS 1936-1971

JOSEPH R. JORDAN
C. RICHARD EBY

WILLIAM R. ALLEN, Ph.D.
JOHN PAUL DAVIS
DOUGLAS A. SCHOLER
BRETT A. SCHATZ
DAVID W. DORTON
SARAH OTTE GRABER
STEVEN W. BENINTENDI, Ph.D.
RANDALL S. JACKSON, JR.

OF COUNSEL
JOHN D. POFFENBERGER
DAVID J. JOSEPHIC
THOMAS W. FLYNN
J. DWIGHT POFFENBERGER, JR.
BRADLEY D. BECK
DONALD F. FREI

January 23, 2006

FACSIMILE COVER SHEET

To: U.S. Patent and Trademark Office
Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22213-1450

Fax: 571-273-8300

Enclosures:
- Fax Cover Sheet containing Certificate of Facsimile Transmission (1 page)
- Transmittal containing Certificate of Facsimile Transmission with authorization to charge Deposit Account 23-3000 in the amount of $500.00 for the Filing Fee (2 pages)
- Appeal Brief (26 total pages, including cover sheet, 19 pages Appeal Brief and 6 pages Claims Appendix)

From: Scott A. Stinebruner
Reg. No. 38,323

Re: U.S. Patent Application

Serial No.	09/975,442
Filed:	October 11, 2001
Applicant:	Jennifer Anne Dervin et al.
Art Unit:	2142
Confirmation No.:	9166
Our Ref:	IBM/204

Pages: 29 (including cover sheet)

MESSAGE/COMMENTS
OFFICIAL

CERTIFICATE OF FACSIMILE TRANSMISSION

I hereby certify that this correspondence and the enclosures noted herein (29 total pages, including cover sheet) are being transmitted via facsimile transmission to the U.S. Patent and Trademark Office, Mail Stop Appeal Brief Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 at 703-872-9306 on January 23, 2006.

Judith L. Volk — Judith L. Volk

January 23, 2006 — Date

PATENT

ATTY. DOCKET NO. IBM/204
Confirmation No. 9166

CERTIFICATE OF FACSIMILE TRANSMISSION

I hereby certify that this correspondence and the enclosures noted herein (29 total pages, including cover sheet) are being transmitted via facsimile transmission to the U.S. Patent and Trademark Office, Mail Stop Appeal Brief - Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 at 571-273-8300 on January 23, 2006.

Judith L Volk — Judith L. Volk
January 23, 2006 — Date

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant	:	Jennifer Anne Dervin et al.	Art Unit: 2142
Serial No.	:	09/975,442	Examiner: Cheryl M. Reid
Filed	:	October 11, 2001	
For	:	DYNAMIC CLUSTER VERSIONING FOR A GROUP	

Cincinnati, Ohio 45202

January 23, 2006

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

TRANSMITTAL OF APPEAL BRIEF (PATENT APPLICATION-37CFR 191)

1. Transmitted herewith is the APPEAL BRIEF in this application with respect to the Notice of Appeal received by the Office on November 21, 2005. Since January 21, 2005 is a Saturday, the deadline for filing the Appeal Brief extends up to and includes January 23, 2006, and this Appeal Brief is timely filed.

2. **STATUS OF APPLICANT**

This application is on behalf of

XX **other than a small entity**

____ small entity

Verified Statement:

___ attached

___ already filed

3. **FEE FOR FILING APPEAL BRIEF**

Pursuant to 37 CFR 1.17(f) the fee for filing the Appeal Brief is:

____	Small entity	$250.00
XX	**Other than a small entity**	**$500.00**

RECEIVED
CENTRAL FAX CENTER

JAN 2 3 2006

4. **EXTENSION OF TIME**

Applicant petitions for an extension of time under 37 C.F.R. 1.136(a) for the total number of months checked below:

Months		Fee for other than small entity	Fee for small entity
_____	one month	$ 120.00	$ 60.00
_____	two months	..... 450.00	..... 225.00
_____	three months	..... 1,020.00	..... 510.00
_____	four months	..... 1,590.00	..... 795.00
_____	five months	..... 2,160.00	..... 1,080.00

Fee: $_____

If an additional extension of time is required, please consider this a petition therefor.

5. **TOTAL FEE DUE**

The total fee due is:

Appeal Brief Fee **$500.00**
Extension Fee _____

6. **FEE PAYMENT**

____ Attached are checks in the sums of _____ and _____, respectively.

XX Charge fee to Deposit Account No. 23-3000.

7. **FEE DEFICIENCY**

XX Charge any additional extension fee required or credit any overpayment to Deposit Account No. 23-3000.

WOOD, HERRON & EVANS, L.L.P.

By _____
Scott A. Stinebruner
Reg. No. 38,323

2700 Carew Tower
441 Vine Street
Cincinnati, Ohio 45202
(513) 241-2324

Attorney Docket No. IBM/204
Confirmation No. 9166

PATENT

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte Jennifer Anne Dervin, Robert Miller, and Laurie Ann Williams

---

Appeal No. _____
Application No. 09/975,442

APPEAL BRIEF

RECEIVED
CENTRAL FAX CENTER

JAN 2 3 2006

PATENT

IBM/204
Confirmation No. 9166

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	Jennifer Anne Dervin et al.	Art Unit:	2142
Serial No.:	09/975,442	Examiner:	Cheryl M. Reid
Filed:	October 11, 2001	Atty. Docket No.:	IBM/204
For:	DYNAMIC CLUSTER VERSIONING FOR A GROUP		

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

I. REAL PARTY IN INTEREST

This application is assigned to International Business Machines Corporation, of Armonk, New York.

II. RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences.

III. STATUS OF CLAIMS

Claims 1-31 are pending in the Application, each of independent claims 1, 11, 18 and 25 being once amended.. All pending claims stand rejected and are now on appeal.

IV. STATUS OF AMENDMENTS

No amendments have been filed subsequent to final rejection (mailed 08/23/2005).

V. SUMMARY OF CLAIMED SUBJECT MATTER

Applicants' invention is generally directed to an apparatus, program product, and method to update the cluster infrastructure version used by a group resident in a clustered computer system without requiring a shut down of the group during the update. The cluster infrastructure software in individual nodes in the clustered computer system is updated from a first version to a second version (which has different program code from the first version) while the group is maintained in an active state. After the cluster infrastructure software is updated, the group is then notified of the update. In response to the notification, the cluster infrastructure version used by the group is dynamically updated to that of the updated cluster infrastructure software, thus making additional functions supported by the new version of the cluster infrastructure software available for use by all group members (Application, page 5, lines 1-15 and page 14, line 20 to page 15, line 17).

As discussed starting at page 1 of the Application, the invention relates to clustered computer systems, which generally incorporate multiple computers, or nodes, that are networked together to cooperatively perform computer tasks through the presentation of a single system image (Application, page 1, lines 1-5). A principal goal of most clustered computer systems is fault tolerance and high availability, such that any downtime in the cluster as a whole is highly undesirable. The invention addresses this goal within the context of upgrading software in an actively running clustered computer system.

Specifically, clustered computer systems typically handle tasks through the use of individual jobs running on different nodes that are logically grouped together as members of an entity referred to as a "group." (Application, page 2, lines 1-17). Underlying these group members on each node of a clustered computer system is cluster infrastructure software, which is roughly analogous to an operating system on a non-clustered computer system to the extent that where an operating system manages the execution of software applications and provides a programming interface through which applications can invoke various support functions, cluster infrastructure software manages the execution of group members and provides a programming interface through which jobs can invoke various cluster-related support functions (Application, page 2, lines 18-29).

It is often desirable to upgrade cluster infrastructure software from time to time, and in this regard, each release of cluster infrastructure software is usually assigned a "version" that distinguishes the release from prior releases of the software. Upgrades to cluster infrastructure software may be desirable, for example, to provide "bug fixes" that correct errors found in previous versions of the software, as well as to add new support services to the software, e.g., to add new functions and capabilities (Application, page 2, line 30 to page 3, line 6).

Upgrading cluster infrastructure software, however, is more complicated than upgrading other types of software such as an operating system. As noted above, a principal goal of clustering is high availability. Whereas an operating system is typically upgraded by shutting down the computer, installing the new version of the software and restarting the computer, shutting down an entire cluster to accommodate a software upgrade is extremely undesirable. A less intrusive procedure is to shut down individual nodes in a clustered computer system to perform software upgrades, while allowing other nodes to remain active and handle system tasks in the absence of any shut down nodes (Application, page 4, lines 5-20).

Due to the interrelated nature of members of a group, however, upgrading individual nodes one at a time does present a number of problems. In particular, during the upgrade process, different nodes will typically have different versions of the cluster infrastructure software running at the same time. As noted above, upgrades to cluster infrastructure software may add new services or functions, some of which may be used by group members after being upgraded. However, to ensure that group members run in a coherent and coordinated fashion, conventional clustered computers systems require each member of a group to use the oldest version of the cluster infrastructure software that is present among any of the nodes in the system, which version is selected when the group initially is created, and cannot be changed without shutting down and restarting the group (Application, page 3, lines 18-29).

Shutting down and restarting a group, however, interrupts the availability of the services provided by that group, and is therefore highly undesirable in most clustered environments. It should be noted that shutting down and restarting a group is significantly different from shutting down and restarting individual nodes and/or any group members resident on those nodes, as while individual nodes may leave or join a cluster at any given time, and group members residing

thereon may leave or join their respective groups, most clustered computer systems are configured to still maintain availability of the services provided by the cluster and its groups so long as other nodes remain active (Application, page 4, lines 12-20).

The independent claims at issue in the appeal address this drawback of conventional clustered computer systems by providing the ability to notify a group of an update to cluster infrastructure software in a clustered computer system in a coordinated fashion, such that the cluster infrastructure version used by the group may be dynamically updated to that of the updated cluster infrastructure software (Application, page 5, lines 1-7).

Fig. 3, and the text at page 12, lines 9-18 of the Application, describe a sequence of operations for dynamically updating the cluster infrastructure software in a clustered computer system. As shown in block 42, the cluster infrastructure software resident on each node in the cluster is individually updated while maintaining the cluster, and all groups thereon, active, using existing node leave and restart functionality, as well as software update functionality. Once the cluster infrastructure software on all nodes has been updated, an adjust version request is sent to any node in the group, as shown in block 44. Once the request has been sent, the cluster infrastructure software version used by each group on each node in the cluster is dynamically updated in block 46.

While other mechanisms may be used to perform the notification of a group, one such mechanism is described in connection with blocks 64-82 of Fig.4, specifically at page 13, line 30 to page 14, line 12, where a "membership change" message is ultimately sent to each member of a group after the cluster infrastructure software has been updated on all of the nodes of the clustered computer system. Once received, each group member processes the message, detects the reason for the message is to update the cluster infrastructure software version, and processes the request to update the version while maintaining the group in an active status. Due to the ordered messaging protocol used, coherency is maintained as other tasks are prohibited from being performed by each group member until the membership change message has been processed (Application, page, 14, lines 13-17).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

A. Claim 18 stands rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter.

B. Claims 1-3, 5-8 and 10[1] stand rejected under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,453,468 to *D'Souza* (*D'Souza*), U.S. Patent No. 6,070,012 to *Eitner et al.* (*Eitner*), and U.S. Patent No. 6,163,855 to *Shrivastava et al.* (*Shrivastava*).

C. Claim 9 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over *D'Souza, Eitner,* and *Shrivastava,* and further in view of U.S. Patent No. 6,769,008 to *Kumar et al.* (*Kumar*).

D. Claims 4 and 5 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *D'Souza, Eitner,* and *Shrivastava*, and further in view of U.S. Patent No. 6,505,257 to *Murata et al.* (*Murata*).

E. Claims 11-13, 15, 18-20 and 22 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *Shrivastava* and *D'Souza.*

F. Claims 16-17 and 23-24 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *Shrivastava* and *D'Souza, Kumar*, and further in view of U.S. Patent No. 5,974,429 to *Strub* (*Strub*).

G. Claims 14 and 21 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *D'Souza* and *Shrivastava* and further in view of *Murata.*

---

[1] Claim 5 is listed in both of Paragraphs 5 and 15 rejections of the Final Office Action, but is discussed more fully by the Examiner in Paragraph 17. Therefore, Applicants assume this is a typographical error and the rejection should refer to claims 1-3, 6-8 and 10.

H. Claims 25-29 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *Shrivastava*, *Murata* and *Eitner*, and further in view of *D'Souza*.

I. Claims 30-31 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over *Shrivastava*, *Murata*, *Eitner* and *D'Souza*, and further in view of *Kumar* and *Strub*.

VII. ARGUMENT

Applicants remarks in rebuttal to the Examiner's rejections are presented below. In some cases, specific discussions of particular claims are not made in the interests of streamlining the appeal. The omission of a discussion with respect to any particular claim, however, should not be interpreted as an acquiescence as to the merits of the Examiner's rejection of the claim, particularly with respect to claims reciting features that are addressed in connection with the rejections applied to other claims pending in the appeal.

A. Claim 18 was improperly rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter.

Claim 18 was rejected as being directed to non-statutory subject matter, for allegedly not being limited to tangible embodiments. Specifically, the Examiner apparently considers the recitation of a signal bearing medium that is described in the specification as including transmission media such as digital or analog communication links (Application, pages 11-12) to incorporate intangible embodiments.

Applicants respectfully submit, however, that there is no requirement in 35 U.S.C. §101, or within any case law of which Applicants are aware, that precludes signal or carrier wave type claims, much less claims that cover "intangible embodiments" as has been recently asserted by the Office. Accordingly, Applicants respectfully request reversal of the Examiner's rejection.

B. Claims 1-3, 6-8[2] and 10 were improperly rejected under 35 U.S.C. § 103(a) as being unpatentable over *D'Souza*, *Eitner*, and *Shrivastava*.

Claims 1-3, 6-8 and 10, of which claim 1 is independent, were rejected as being unpatentable over *D'Souza*, *Eitner*, and *Shrivastava*. A *prima facie* showing of obviousness requires that the Examiner establish that the differences between a claimed invention and the prior art "are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art." 35 U.S.C. §103(a). Such a showing requires that all claimed features be disclosed or suggested by the prior art. Such a showing also requires objective evidence of the suggestion, teaching or motivation to combine or modify prior art references, as "[c]ombining prior art references without evidence of such a suggestion, teaching or motivation simply takes the inventor's disclosure as a blueprint for piecing together the prior art to defeat patentability -- the essence of hindsight." In re Dembiczak, 50 USPQ2d 1614, 1617 (Fed. Cir. 1999)."

Applicant respectfully submits that, in the instant case, the Examiner has failed to establish a *prima facie* case of obviousness as to claim 1, and as such, the rejection thereof should be reversed.

Claim 1 generally recites a method of updating a cluster infrastructure version used by a group resident in a clustered computer system of the type including a plurality of nodes. The method includes updating the cluster infrastructure software in individual nodes in the clustered computer system while the group is maintained in an active state. After the cluster infrastructure software is updated, the group is notified of the update to the cluster infrastructure software, and in response to the notification, a cluster infrastructure version used by the group to that of the updated cluster infrastructure software is dynamically updated. Of note, the update of the cluster infrastructure software is from a first version to a second version, where the second version of the cluster infrastructure software has different program code from the first version of the cluster infrastructure software.

---

[2] See Footnote 1.

As discussed, for example, at page 2, lines 18-29, cluster infrastructure software is analogous in many respects to an "operating system" for a clustered computer system, providing a number of services such as managing the execution of members of a cluster group, providing a programming interface for jobs to invoke cluster-related support functions, etc. Applicants' invention is directed to dynamically updating this type of software in an active clustered computer system, with minimal interruption of availability in the system. As noted, for example at page 14, lines 20-25, updates to cluster infrastructure software may be performed, for example, to fix bugs in the software and/or to add new features or functions. Fig. 2 of the Application, as well as the supporting text at page 9, lines 28-30, distinctly distinguish between jobs or applications and cluster infrastructure software ("One or more jobs or applications 34 are also illustrated in node 10, each having access to features implemented within the cluster infrastructure software 30.")

In rejecting claim 1, the Examiner relies on *D'Souza*, *Eitner* and *Shrivastava*. Of note, however, none of these references even address performing updates to cluster infrastructure software, much less the specific notification and dynamic update protocol recited in claim 1. Instead, it appears that the rejection of claim 1 is highly reliant on hindsight for its support.

D'Souza discloses updating applications in a clustered environment; however, it is apparent from the reference that the underlying cluster infrastructure software is not updated. Instead, *D'Souza* discloses the update of "business logic software modules," which are more analogous to members of a cluster group, if at all. In rejecting claim 1, the Examiner relies on col. 7, lines 20-25, arguing that *D'Souza* discloses updating cluster infrastructure software from a first version to a second version. However, the cited passage refers to upgrading a software program implemented as "software modules" running on clustered computers. From a review of the reference, it should be apparent that "software modules" do not constitute the cluster infrastructure software executed by a computer in a cluster. Col. 9, lines 30-38 states as follows:

> [T]he servers within each stage and within each cluster may be heterogeneous (i.e., implemented on different platforms and having different capability) and each may operate a different set of business logic modules, i.e., application software modules. By way

> of example, servers 216, 218, 220 and 222 within business logic stage 206 may be implemented using different hardware/software platforms and configurations that are adapted for operating the business logic software modules implemented therein. (*emphasis added*).

In this passage *D'Souza* expressly notes that the software/business logic modules are required to be "application" software modules (note, for example, the usage of "i.e." instead of "e.g."). Moreover, the passage notes that different "hardware/software platforms" can be adapted for operating those software modules. One of ordinary skill in the art would readily appreciate that "cluster infrastructure software" constitutes a portion of the "software platform" utilized by a server operating in a clustered environment. Given that *D'Souza* expressly distinguishes the business logic software modules from the software platform that operates those modules, it is readily apparent that the "business logic software modules" of *D'Souza* are not analogous to cluster infrastructure software. It is also important to note that nowhere in *D'Souza* is there any detailed discussion of upgrading an operating system or any particular software platform for a cluster node; it is assumed in *D'Souza* that such software is resident on each node, and no details of the operation or configuration of such software is even ever provided in *D'Souza*.

Consequently, it is readily apparent that *D'Souza* does not disclose updating cluster infrastructure software as asserted by the Examiner.

The Examiner next relies on *Eitner* for allegedly teaching the updating of software while a group is maintained in an active state. However, *Eitner* discloses the updating of software in a PBX system where a switchover occurs in a single component after upgraded software is installed in a different portion of the component's memory. The passage cited by the Examiner, at col. 4, lines 49-53, merely describes "hot" downloading software into one bank of a component's memory while another version of the software is running in another bank of the same component's memory, which is then activated by swapping base addresses in the component.

Importantly, the term "cluster" is not found anywhere in the reference, which is not surprising considering its application in a PBX system used in the telecommunications field. The

reference does not disclose any type of interaction between Network System (NS) devices, or any coordination between that is even remotely analogous to the interaction of nodes, and the members of a group resident thereon, in a clustered computer system. There is no appreciation in the reference for the concept of a cluster group, much less the maintenance of a group in an active state during the update of software in a cluster node. The concept of a clustered computer system, as well as the concept of a group that is distributed amongst multiple nodes in a clustered computer system, has a distinct meaning in the art, and is explicitly defined in the application, e.g., at page 1, line 1 to page 2, line 17. The Examiner has apparently chosen to disregard this lexicographical material, and generally interpret the concept of a cluster as any collection of networked devices, and a group as any collection of software, irrespective of how (or even whether) such devices and/or software interact with one another. Doing so, however, effectively reads these terms out of the claim altogether, and is thus improper.

Given also that *Eitner*, like *D'Souza* does not teach updating cluster infrastructure software, Applicants respectfully submit that *Eitner* adds nothing to the Examiner's rejection.

Shrivastava likewise adds little, if anything, to the Examiner's rejection. The Examiner apparently relies on *Shrivastava* for teaching the notification of a group of an update to cluster infrastructure software, along with dynamically updating a cluster infrastructure version used by the group, citing col. 6, lines 27-42. *Shrivastava*, however, merely discloses the propagation of cluster configuration changes among nodes in a cluster. Indeed, the reference discloses software that is reasonably analogous to cluster infrastructure software, including a number of the services incorporated into cluster service 70 illustrated in Fig. 3 and described in cols. 5 and 6. Notably, however, the "updates" that are performed in the reference are to a cluster configuration database 82, and not to the software in cluster service 70. The passage cited by the Examiner, in particular, discloses an event processor that processes transactions to update configuration data in the cluster. Neither the passage, nor the remainder of the reference, addresses updating software running on a node in the cluster.

It is also important to note that claim 1 was amended during prosecution to clarify that the update of the cluster infrastructure software from the first version to the second version incorporates a change in the program code from the first to the second version. Consequently,

the update of cluster configuration data disclosed in *Shrivastava* falls far short of disclosing the update to the cluster infrastructure software recited in claim 1.

As such, given that *Shrivastava* does not disclose an update to the cluster infrastructure software, *Shrivastava* cannot be read to disclose "notifying [a] group of [an] update to the cluster infrastructure software," as has been asserted by the Examiner. In fact, Applicants cannot even find any notification made specifically to a group in *Shrivastava*, much less a notification made to a group that specifically addresses the fact that cluster infrastructure software was updated.

Shrivastava similarly does not disclose the concept of dynamically updating a cluster infrastructure version used by [a] group to that of the updated cluster infrastructure software," as also required by claim 1. Indeed, the concept of a "version" is not addressed anywhere in *Shrivastava*, and the reference, similarly to *D'Souza* and *Eitner*, does not even appreciate the concept that a group might be configurable to "use" different versions of underlying software. As discussed, for example, at page 14, lines 20-25, enabling a group to "use" a particular version of software could permit a group to access a new function made available in a new version of cluster infrastructure software.

Applicants therefore respectfully submit that none of the cited references disclose or suggest a number of concepts recited in claim 1. First, none of the references disclose or suggest updating cluster infrastructure software in a clustered computer system. Second, none of the references disclose or suggest notifying a group in a clustered computer system of an update made to cluster infrastructure software. Third, none of the references disclose or suggest dynamically updating a cluster infrastructure version used by a group to that of updated cluster infrastructure software.

Applicants respectfully submit that the rejection is replete with hindsight-based analysis, and is thus deficient on its face. It seems the approach taken in the rejection is to generally find a reference disclosing an update of software being made in a clustered computer system, a reference disclosing an update of software being made without requiring some electronic device to be turned off, and a reference disclosing the notification of some entity in a clustered computer system of a configuration change being made in the system. These disclosures, however, cannot be combined together to specifically teach the combination of features in claim 1 without filling

in the blanks with hindsight-based reasoning. None of the references addresses the problems associated withe updating cluster infrastructure software in a dynamic manner without compromising system availability. None of the references addresses the notification of a group of an update to the cluster infrastructure software over which the group executes. None of the references even addresses controlling what version of cluster infrastructure software a group is configured to use. In addition, the Examiner has provided no specific evidence as to why one of ordinary skill in the art would be motivated by the prior art to modify *D'Souza* to incorporate the specific features alleged to correspond to claim 1. A general motivation to combine the references, as asserted by the Examiner, still fails to support a motivation to incorporate the specific features recited in claim 1 to the *D'Souza* system. It is only through the benefit of hindsight that these specific concepts can be overlaid onto the teachings of the references.

Accordingly, Applicants submit that the Examiner has failed to raise a *prima facie* case of obviousness as to claim 1. Applicants therefore respectfully submit that a clear error exists with respect to the Examiner's rejection of claim 1, and that the rejection of this claim, and of claims 2-3, 6-8 and 10 which depend therefrom, should be reversed, and the claims allowed over the prior art of record.

C. Claim 9 was improperly rejected under 35 U.S.C. § 103(a) as being unpatentable over *D'Souza*, *Eitner*, *Shrivastava*, and further in view of *Kumar*.

Claim 9 depends from claim 1, and additionally recites the further step of "verifying that the group is not partitioned prior to notifying the group." In rejecting claim 9, the Examiner takes the position that *Kumar* discloses preventing partitioning and that *Shrivastava* discloses notifying groups. This rejection, unfortunately, is indicative of many of the other rejections set forth in the Office Action in terms of parsing claim language to effectively destroy the overall meaning of the language as a whole. While *Kumar* may disclose preventing partitioning, the reference does not disclose "verifying" whether a group is partitioned, nor does the reference disclose the specific temporal relationship of performing the verification "prior to" notifying a group. Furthermore, as discussed above in connection with claim 1, *Shrivastava* does not even

disclose specifically notifying a group. As such, Applicants submit that the rejection of claim 9 is not proper, and should be reversed. Allowance of claim 9 is therefore respectfully requested.

D. Claims 4-5 were improperly rejected under 35 U.S.C. § 103(a) as being unpatentable over *D'Souza, Eitner, Shrivastava*, and further in view of *Murata*.

Claims 4 and 5 depend from claim 1, which as discussed above, is patentable over *D'Souza*, *Eitner* and *Shrivastava*. Furthermore, *Murata* adds nothing to the rejection of claim 1. Accordingly, claims 4 and 5 are patentable by virtue of their dependency upon claim 1. Reversal of the Examiner's rejections, and allowance of these claims, are therefore respectfully requested.

E. Claims 11-13, 15, 18-20 and 22 were improperly rejected under 35 U.S.C. § 103(a) as being unpatentable over *Shrivastava* and *D'Souza*.

Independent claims 11 and 18 each recite program code resident in a node of a clustered computer system that is configured to notify a member that is also resident on the node of an update to the cluster infrastructure software on the node from a first version to a second version, where the second version of the cluster infrastructure software has different program code from the first version of the cluster infrastructure software. Each claim also recites that the program code is configured to dynamically update the cluster infrastructure version used by the member to that of the updated cluster infrastructure software.

In rejecting these claims, the Examiner relies on *Shrivastava* and *D'Souza*. However, as discussed above in connection with claim 1, these references fail to disclose a number of the features recited in these claims. In particular, claims 11 and 18 recite the notification of a group member of an update to cluster infrastructure software on the same node. As noted above, however, *Shrivastava* discloses only updates to a cluster configuration database, and does not discuss updates to the program code in cluster infrastructure software. Furthermore, there is no disclosure in *Shrivastava* of any notification that is specifically made to a group member, much less such a notification that is specifically for the purpose of providing a notification that an update has occurred to the cluster infrastructure software. The cited passage, at col. 6, lines 19-

41, merely discloses an event processor that processes transactions to update cluster configuration data.

Claims 11 and 18 also recite dynamically updating the cluster infrastructure version used by a member of a group to that of updated cluster infrastructure software. The cited passage at col. 6, lines 19-41, as well as the rest of the reference, is utterly silent with respect to any particular version of software being "used by" a member of a group, much less updating such a version to that of updated cluster infrastructure software.

D'Souza is cited merely for disclosing the concept of first and second versions of software. As discussed above in connection with claim 1, however, *D'Souza* fails to even disclose updates to cluster infrastructure software, so this reference adds little, if anything, to the rejection.

Accordingly, Applicants respectfully submit that the Examiner has failed to raise a *prima facie* case of obviousness with respect to claims 11 and 18. Reversal of the rejections, and allowance of claims 11 and 18, and of claims 12-13, 15, 19-20 and 22 which depend therefrom, are therefore respectfully requested.

F. Claims 16-17 and 23-24 were improperly rejected under 35 U.S.C. § 103(a) as being unpatentable over *Shrivastava*, *D'Souza*, and *Kumar* and further in view of *Strub*.

Claims 16 and 23

Claims 16 and 23 respectively depend from claims 11 and 18, and additionally recite the concept of verifying that a group is not partitioned, and if partitioned, returning an error message. In rejecting claims 16 and 23, the Examiner refers to the rejection of claim 9, which relies on *Kumar* for allegedly disclosing preventing partitioning. As noted above in connection with claim 9, *Kumar* does not disclose "verifying" whether a group is partitioned. Furthermore, the Examiner relies on *Strub* for allegedly disclosing an error message, but the cited passage at col. 7, lines 23-25 refers to an error message associated with an inability to propagate a journal. Applicants are not attempting to claim all error messages associated with clustering errors, and as such, the Examiner's citation of a reference that merely discloses an error message generated in a

clustered computer system for an entirely different error falls far short of disclosing the specific error message recited in claims 16 and 23. As such, Applicants submit that the rejections of claims 16 and 23 are not proper, and should be reversed. Allowance of claims 16 and 23 is therefore respectfully requested.

Claims 17 and 24

Claims 17 and 24 respectively depend from claims 11 and 18, and additionally recite the concept of determining whether a node is capable of running updated cluster infrastructure software prior to notifying a member, and if not capable, returning an error message. In rejecting these claims, the Examiner refers to the rejection of claim 10, which relies on *D'Souza*, col. 3, lines 39-52 for allegedly disclosing verifying whether nodes are capable of running updated software. However, it should be noted that the cited passage merely refers to compatibility of software, not to any specific step performed to verify compatibility. Thus, the Examiner has failed to establish that *D'Souza* discloses verifying whether a node is capable of running updated cluster infrastructure software, as required by these claims.

Furthermore, the Examiner relies on *Strub* for allegedly disclosing an error message, but the cited passage at col. 7, lines 23-25 refers to an error message associated with an inability to propagate a journal. Applicants are not attempting to claim all error messages associated with clustering errors, and as such, the Examiner's citation of a reference that merely discloses an error message generated in a clustered computer system for an entirely different error falls far short of disclosing the specific error message recited in claims 17 and 24. As such, Applicants submit that the rejections of claims 17 and 24 are not proper, and should be reversed. Allowance of claims 17 and 24 is therefore respectfully requested.

G. Claims 14 and 21 were improperly rejected under 35 U.S.C. § 103(a) as being unpatentable over *D'Souza* and *Shrivastava* and further in view of *Murata*.

Claims 14 and 21 depend from claims 11 and 18, which as discussed above, are patentable over *D'Souza* and *Shrivastava*. Furthermore, *Murata* adds nothing to the rejections of claims 11 and 18. Accordingly, claims 14 and 21 are patentable by virtue of their dependency upon claims 11 and 18. In addition, these claims additionally recite the concept of using a membership change message with an adjust version reason code to notify a member of a group of an update to cluster infrastructure software. The Examiner refers to the rejection of claim 7; however, this rejection does not even apply *Murata*, and instead relies on *Shrivastava*, and in particular the cited passages at col. 5, lines 27-38 and col. 6, lines 42-60. These passages generally disclose cluster membership, but fall far short of disclosing the specific use of a membership change message, much less one with the recited reason code, for the purpose of specifically notifying a group of an update to cluster infrastructure software. Reversal of the Examiner's rejections, and allowance of these claims, are therefore respectfully requested.

H. Claims 25-29 were improperly rejected under 35 U.S.C. § 103(a) as being unpatentable over *Shrivastava*, *Murata* and *Eitner*, and further in view of *D'Souza*.

Independent claim 25 generally recites a cluster computer system that includes a plurality of nodes, each having resident thereon cluster infrastructure software, a group including a plurality of group members resident on the plurality of individual nodes, and program code resident on the plurality of nodes. The program code is configured to shutdown and restart individual nodes among the plurality of nodes while maintaining the group in an active state so that the cluster infrastructure software resident on such individual nodes can be updated to incorporate different program code while such individual nodes are shutdown. The program code is also configured to notify the group of the update to the cluster infrastructure software after the cluster infrastructure software has been updated in each of the plurality of nodes, and to dynamically update a cluster infrastructure version used by the group to that of the updated cluster infrastructure software.

Claim 25 is rejected as being obvious in light of *Shrivastava*, *Murata*, *Eitner* and *D'Souza*. As noted above in connection with claims 1, 11 and 18, however, none of *Shrivastava*, *Eitner* and *D'Souza* disclose any change in a cluster involving a change in program code in cluster infrastructure software. None of the references discloses or suggests a dynamic mechanism for updating cluster infrastructure software to incorporate different program code in the manner recited in the claim, specifically the notification of a group of an update to cluster infrastructure software, or the dynamic update of the cluster infrastructure version used by a group to that of updated cluster infrastructure software.

Murata is similarly deficient with respect to the aforementioned limitations, and as such, these limitations are distinguishable from the art cited by the Examiner.

Claim 25 additionally recites that the program code is configured to shut down and restart individual nodes while maintaining a group in an active state so that the cluster infrastructure software on each individual node can be updated to incorporate different program code. For this feature, the Examiner relies on *Murata*, and in particular, col. 1, lines 57-60 and col. 2, lines 1-7. These passages, however, refer to inhibiting the acceptance of jobs for a processor in a multiprocessor system. In this regard, the reference utilizes a different usage of the term "cluster", as it is used in the reference to refer to a logical grouping of processors in a multiprocessor system. Within the context of the invention, where a cluster is a group of computers that are clustered together to present a single system image, *Murata* and its reference to inhibiting acceptance of jobs is entirely irrelevant to the concept of starting or shutting down a node in a clustered computer system.

As such, Applicants respectfully submit that the Examiner has failed to raise a *prima facie* case of obviousness with respect to claim 25. Reversal of the rejection, and allowance of claim 25, and of claims 26-29 which depend therefrom, are therefore respectfully requested.

I. Claims 30-31 were improperly rejected under 35 U.S.C. § 103(a) as being unpatentable over *Shrivastava*, *Murata*, *Eitner*, *D'Souza*, and further in view of *Kumar* and *Strub*.

Claim 30

Claims 30 depends from claim 25, and additionally recites the concept of verifying that a group is not partitioned, and if partitioned, returning an error message. In rejecting claim 30, the Examiner refers to the rejection of claim 16, which relies on *Kumar* for allegedly disclosing preventing partitioning. As noted above in connection with claim 16, *Kumar* does not disclose "verifying" whether a group is partitioned. Furthermore, the Examiner apparently relies on *Strub* for allegedly disclosing an error message, but the cited passage at col. 7, lines 23-25 refers to an error message associated with an inability to propagate a journal. Applicants are not attempting to claim all error messages associated with clustering errors, and as such, the Examiner's citation of a reference that merely discloses an error message generated in a clustered computer system for an entirely different error falls far short of disclosing the specific error message recited in claim 30. As such, Applicants submit that the rejection of claim 30 is not proper, and should be reversed. Allowance of claim 30 is therefore respectfully requested.

Claim 31

Claim 31 depends from claim 25, and additionally recite the concept of determining whether a node is capable of running updated cluster infrastructure software prior to notifying a member, and if not capable, returning an error message. In rejecting this claim, the Examiner refers to the rejection of claim 17, which in turn relies on the rejection of claim 10, which relies on *D'Souza*, col. 3, lines 39-52 for allegedly disclosing verifying whether nodes are capable of running updated software. However, it should be noted that the cited passage merely refers to compatibility of software, not to any specific step performed to verify compatibility. Thus, the Examiner has failed to establish that *D'Souza* discloses verifying whether a node is capable of running updated cluster infrastructure software, as required by this claim.

Furthermore, the Examiner relies on *Strub* for allegedly disclosing an error message, but the cited passage at col. 7, lines 23-25 refers to an error message associated with an inability to

propagate a journal. Applicants are not attempting to claim all error messages associated with clustering errors, and as such, the Examiner's citation of a reference that merely discloses an error message generated in a clustered computer system for an entirely different error falls far short of disclosing the specific error message recited in claim 31. As such, Applicants submit that the rejection of claim 31 is not proper, and should be reversed. Allowance of claim 31 is therefore respectfully requested.

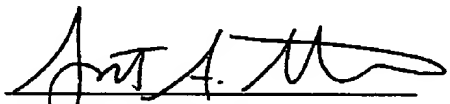
VIII. CONCLUSION

In conclusion, Applicant respectfully requests that the Board reverse the Examiner's rejections of claims 1-31, and that the Application be passed to issue. If there are any questions regarding the foregoing, please contact the undersigned at 513/241-2324. Moreover, if any other charges or credits are necessary to complete this communication, please apply them to Deposit Account 23-3000.

Respectfully submitted,

WOOD, HERRON & EVANS, L.L.P.

Date: 23 JAN 2006

By: 

Scott A. Stinebruner
Reg. No. 38,323

2700 Carew Tower
441 Vine Street
Cincinnati, Ohio 45202
(513) 241-2324 - Telephone
(513) 241-6234 - Facsimile

IX. CLAIMS APPENDIX: CLAIMS ON APPEAL (S/N 09/975,442)

1. (Once Amended) A method of updating a cluster infrastructure version used by a group resident in a clustered computer system of the type including a plurality of nodes, the method comprising:

(A) updating the cluster infrastructure software from a first version to a second version in individual nodes in the clustered computer system while the group is maintained in an active state, wherein the second version of the cluster infrastructure software has different program code from the first version of the cluster infrastructure software;

(B) after the cluster infrastructure software is updated, notifying the group of the update to the cluster infrastructure software; and,

(C) in response to the notification, dynamically updating a cluster infrastructure version used by the group to that of the updated cluster infrastructure software.

2. (Original) The method of claim 1, wherein the updated cluster infrastructure software includes at least one new function, whereby the group has access to the new function subsequent to dynamically updating the cluster infrastructure version used by the group.

3. (Original) The method of claim 1, further comprising notifying all groups resident in the clustered computer system after the cluster infrastructure software is updated.

4. (Original) The method of claim 1, wherein updating the cluster infrastructure software in an individual node comprises shutting down the node, installing cluster infrastructure software on the node, and restarting the node.

5. (Original) The method of claim 4, wherein shutting down the node includes removing a member that is resident on the node from the group and wherein restarting the node includes adding the member to the group.

6. (Original) The method of claim 1, wherein notifying comprises sending a ordered message to the group.

7. (Original) The method of claim 6, wherein notifying comprises sending a membership change message with an adjust version reason code.

8. (Original) The method of claim 1, further comprising verifying that all nodes are active prior to notifying the group.

9. (Original) The method of claim 1, further comprising verifying that the group is not partitioned prior to notifying the group.

10. (Original) The method of claim 1, further comprising verifying that all nodes are capable of running the updated cluster infrastructure version prior to notifying the group.

11. (Once Amended) An apparatus comprising:

(A) a node configured to participate in a clustered computer system, the node having resident thereon cluster infrastructure software and at least one member of a group; and,

(B) program code resident in the node, the program code configured to notify the member of an update to the cluster infrastructure software from a first version to a second version, and to dynamically update a cluster infrastructure version used by the member to that of the updated cluster infrastructure software; wherein the second version of the cluster infrastructure software has different program code from the first version of the cluster infrastructure software.

12. (Original) The apparatus of claim 11, wherein the updated cluster infrastructure software includes at least one new function, whereby the group has access to the new function subsequent to dynamically updating the cluster infrastructure version used by the node.

13. (Original) The apparatus of claim 11, wherein the notification is made using ordered messaging.

14. (Original) The apparatus of claim 13, wherein the notification is made via a membership change message with an adjust version reason code.

15. (Original) The apparatus of claim 11, wherein the program code is further configured to verify that the node is active prior to notifying the member and, if the node is not active, to return an error message.

16. (Original) The apparatus of claim 11, wherein the program code is further configured to verify that the group is not partitioned prior to notifying the member and, if the group is partitioned, to return an error message.

17. (Original) The apparatus of claim 11, wherein the program code is further configured to determine whether the node is capable of running the updated cluster infrastructure software prior to notifying the member and, if the node is not capable of running the updated cluster infrastructure software, to return an error message.

18. (Once Amended) A program product, comprising:

(A) program code configured to reside on a node that participates in a clustered computer system and that further has resident thereon cluster infrastructure software and at least one member of a group, the program code configured to notify the member of an update to the cluster infrastructure software from a first version to a second version, and to dynamically update a cluster infrastructure version used by the member to that of the updated cluster infrastructure software; and,

(B) a signal-bearing medium bearing the program code;

wherein the second version of the cluster infrastructure software has different program code from the first version of the cluster infrastructure software.

19. (Original) The program product of claim 18, wherein the updated cluster infrastructure software includes at least one new function, whereby the group has access to the new function subsequent to dynamically updating the cluster infrastructure version used by the node.

20. (Original) The program product of claim 18, wherein the notification is made using ordered messaging.

21. (Original) The program product of claim 20, wherein the notification is made via a membership change message with an adjust version reason code.

22. (Original) The program product of claim 18, wherein the program code is further configured to verify that the node is active prior to notifying the member and, if the node is not active, to return an error message.

23. (Original) The program product of claim 18, wherein the program code is further configured to verify that the group is not partitioned prior to notifying the member and, if the group is partitioned, to return an error message.

24. (Original) The program product of claim 18, wherein the program code is further configured to determine whether the node is capable of running the updated cluster infrastructure software prior to notifying the member and, if the node is not capable of running the updated cluster infrastructure software, to return an error message.

25. (Once Amended) A cluster computer system, comprising:

(A) a plurality of nodes, each having resident thereon cluster infrastructure software;

(B) a group including a plurality of group members resident on the plurality of individual nodes; and,

(C) program code resident on the plurality of nodes, the program code configured to shutdown and restart individual nodes among the plurality of nodes while maintaining the group in an active state so that the cluster infrastructure software resident on such individual nodes can be updated to incorporate different program code while such individual nodes are shutdown, the program code further configured to notify the group of the update to the cluster infrastructure software after the cluster infrastructure software has been updated in each of the plurality of nodes, and to dynamically update a cluster infrastructure version used by the group to that of the updated cluster infrastructure software.

26. (Original) The clustered computer system of claim 25, wherein the updated cluster infrastructure software includes at least one new function, whereby the group has access to the new function subsequent to dynamically updating the cluster infrastructure version used by the node.

27. (Original) The clustered computer system of claim 25, wherein the notification is made using ordered messaging.

28. (Original) The clustered computer system of claim 27, wherein the notification is made via a membership change message with an adjust version reason code.

29. (Original) The clustered computer system of claim 25, wherein the program code is further configured to verify that the node is active prior to notifying the member and, if the node is not active, to return an error message.

30. (Original) The clustered computer system of claim 25, wherein the program code is further configured to verify that the group is not partitioned prior to notifying the member and, if the group is partitioned, to return an error message.

31. (Original) The clustered computer system of claim 25, wherein the program code is further configured to determine whether the node is capable of running the updated cluster infrastructure software prior to notifying the member and, if the node is not capable of running the updated cluster infrastructure software, to return an error message.